

Using the Microsoft SQL Server 2012 Management Studio

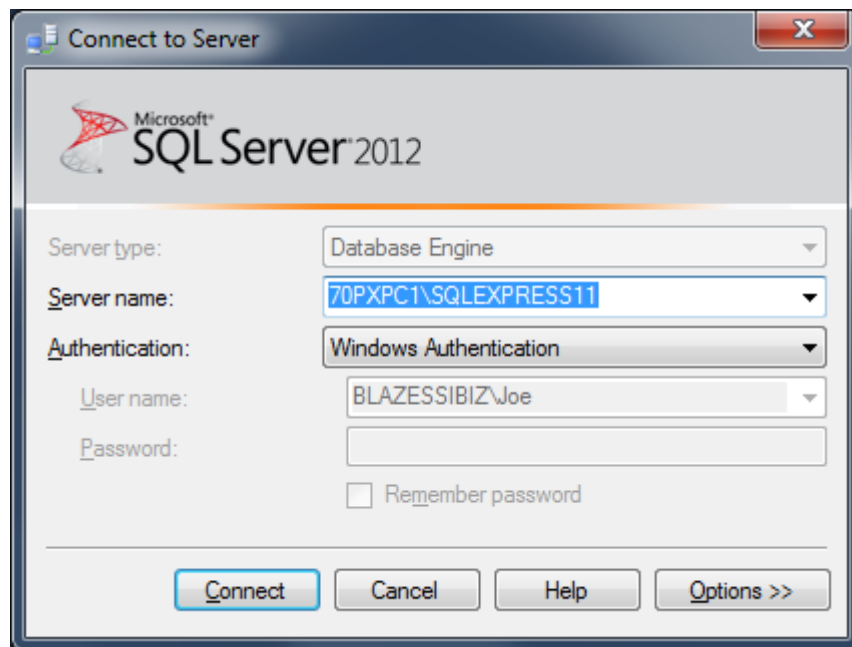
What is the Management Studio?

The tool can be used to manage Microsoft SQL Server instances on a local or remote computer including create/edit/delete databases.

Start It Up:

After installing Microsoft SQL Server 2012 (Express or other version), use the Start > All Programs > Microsoft SQL Server 2012 > SQL Server Management Studio menu item.

In the Connect to Server dialog, select the Server Name and Authentication type for the selected instance of Microsoft SQL Server. If SQL Server Authentication is selected, contact your database administrator for the applicable Login and Password.



If desired, click the Options button to present the Login, Connection Properties and Additional Connection Parameters tabs in the Connect to Server dialog:

Microsoft®
SQL Server® 2012

Login | Connection Properties | Additional Connection Parameters

Server

Type the server name, or choose it from the drop-down list.

Server type: Database Engine

Server name: 70PXP1\SQLEXPRESS11

Authentication: Windows Authentication

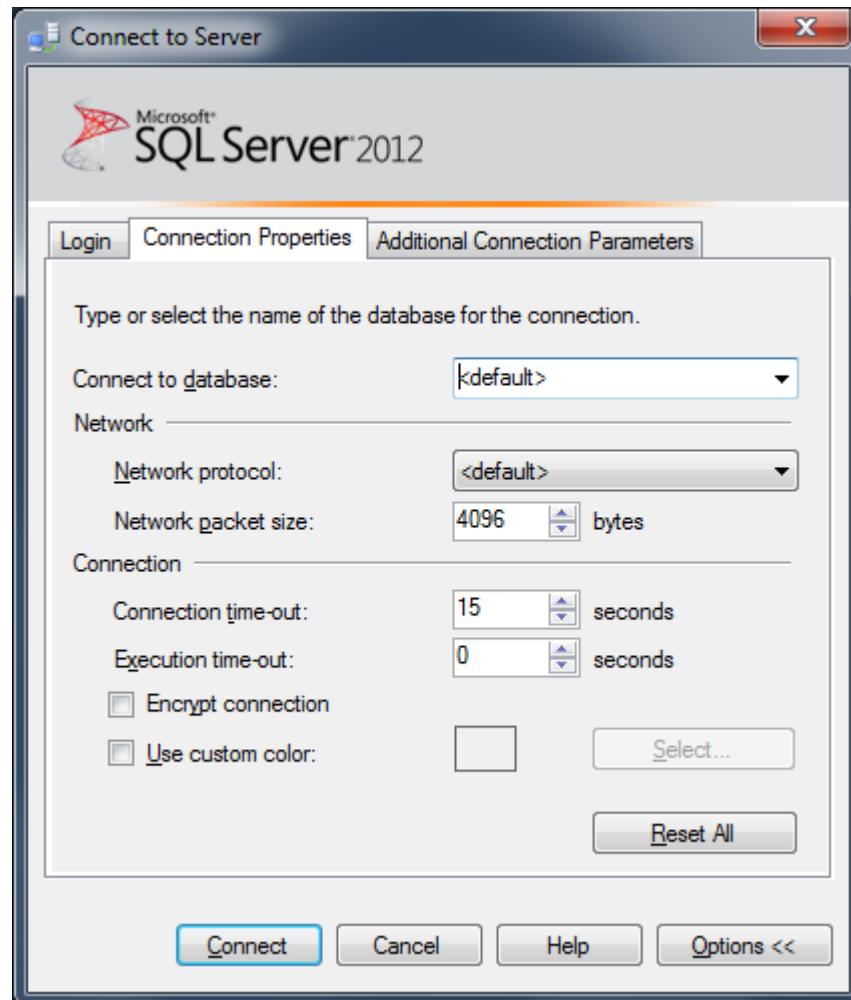
User name: BLAZESSIBIZ\Joe

Password:

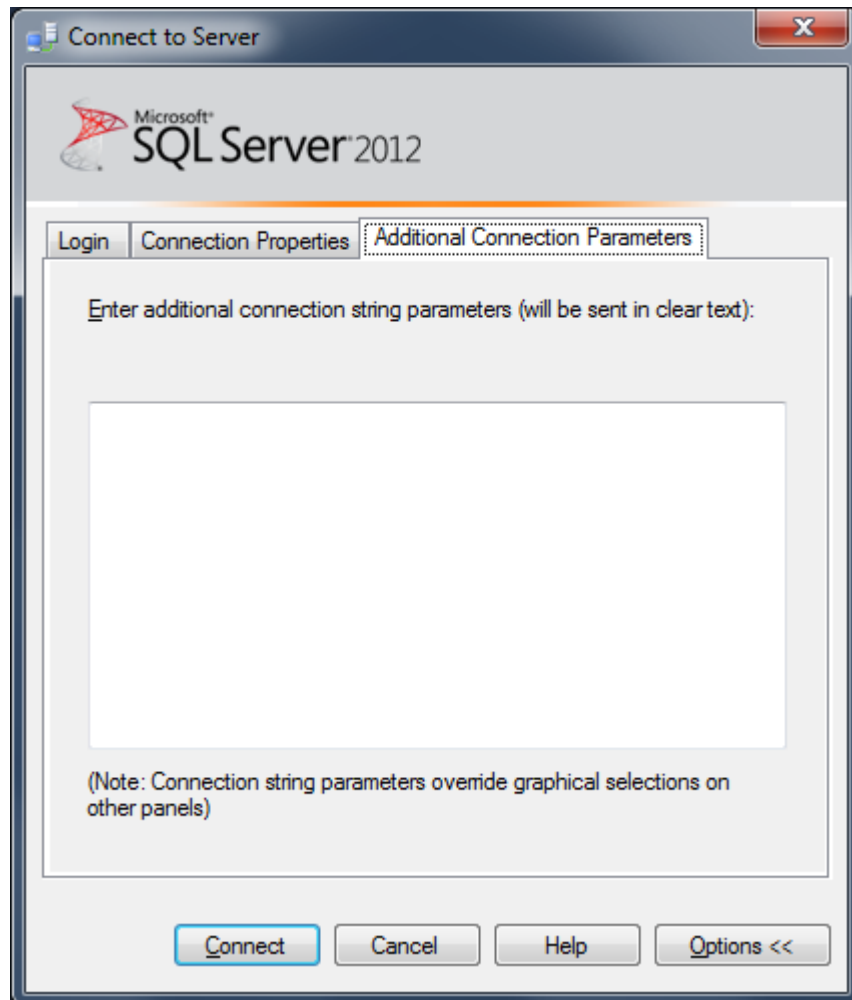
☐ Remember password

Connect Cancel Help Options <<

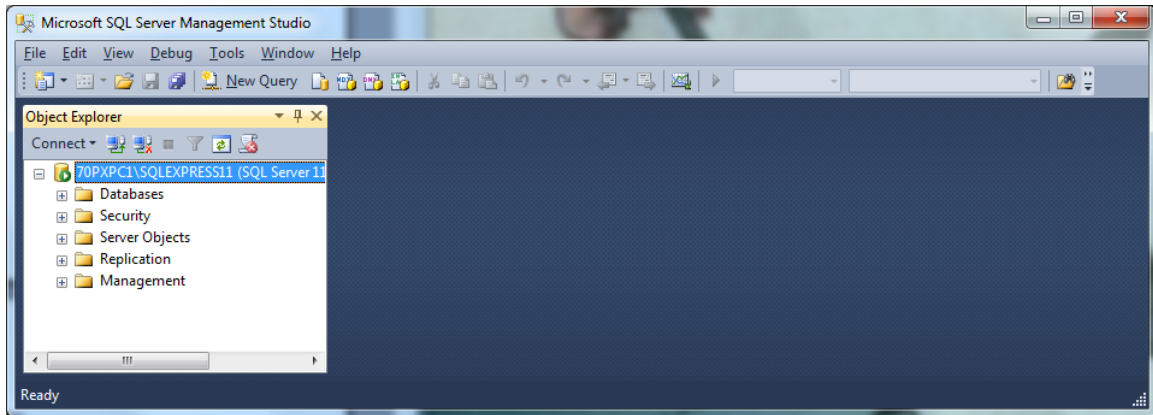
Contact your database administrator to determine how to set the Connection Properties:



Contact your database administrator to determine how to set the Additional Connection Properties:

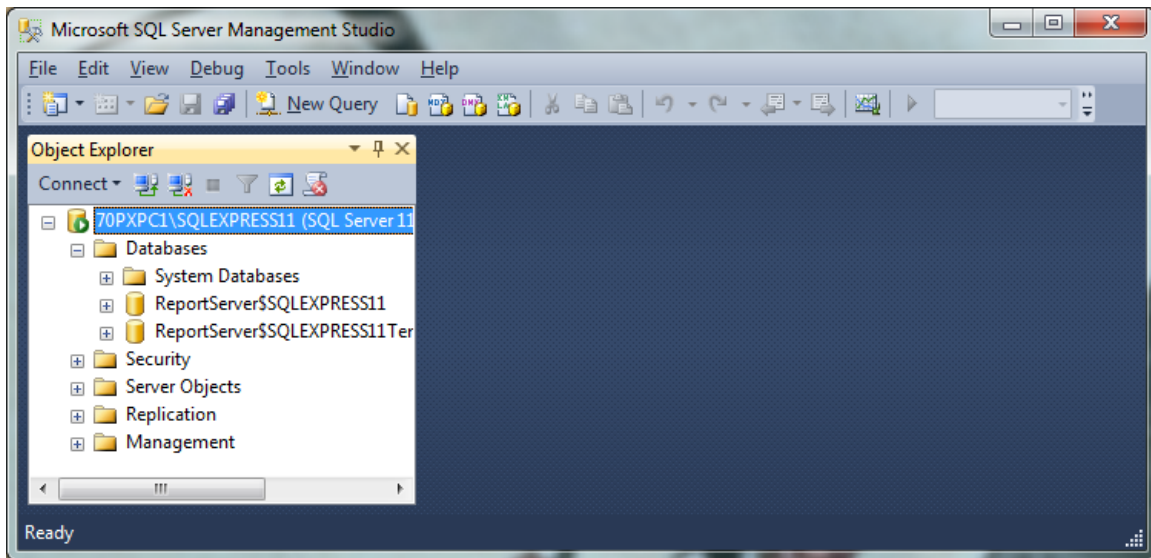


Clicking the Connect button on the Connect to Server dialog will present the Management Studio main window:

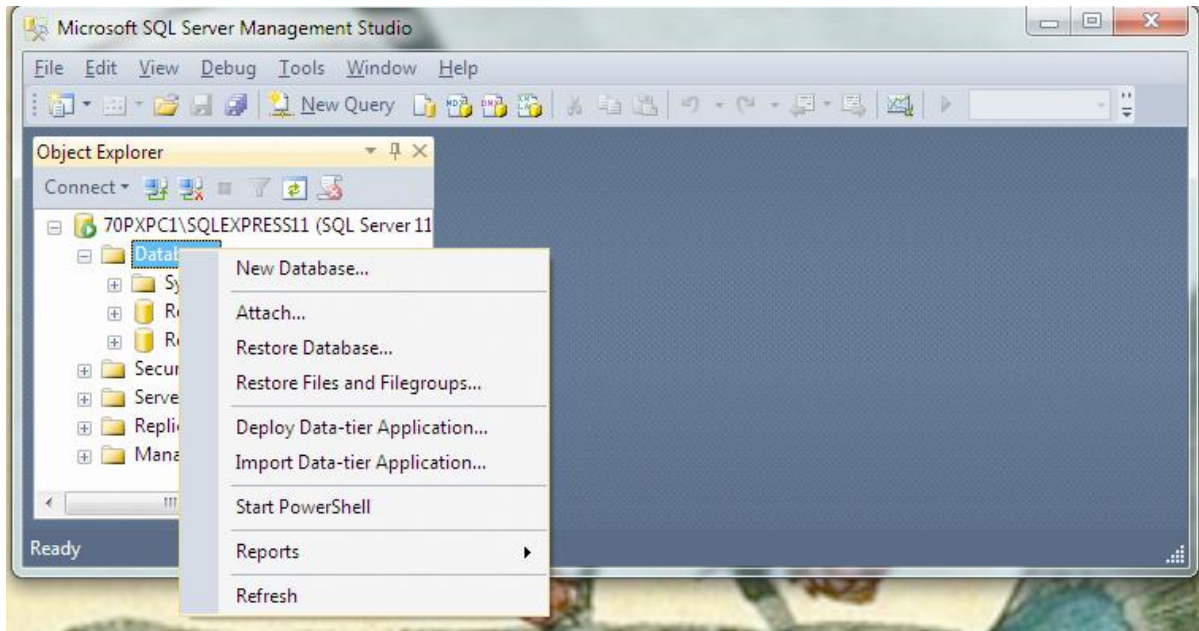


Add a New Database:

Open the Databases tree node:



Right click the Databases tree node and select New Database...:



Enter the Database name (dbOne) and if desired specify the Initial Size and Autogrowth options on the General dialog:

New Database

Select a page: General, Options, Filegroups

Script Help

Database name: dbOne

Owner: <default>

☒ Use full-text indexing

Database files:

Logical Name	File Type	Filegroup	Initial Size (MB)	Autogrowth / Maxsize	Path
dbOne	Rows ...	PRIMARY	4	By 1 MB, Unlimited	c:\Program Files\Microsoft SQL Server\MSSQL11.SQLEXPRESS11\MSSQL\DATA
dbOne_log	Log	Not Applicable	1	By 10 percent, Unlimited	c:\Program Files\Microsoft SQL Server\MSSQL11.SQLEXPRESS11\MSSQL\DATA

Connection:

Server: 70PXP1\SQLEXPRESS11

Connection: BLAZESS161ZJoe

[View connection properties](#)

Progress: Ready

Add Remove OK Cancel

Contact your database administrator to determine the appropriate values to enter on the Optional dialog:

New Database

Select a page: General, **Options**, Filegroups

Script Help

Collation: <default>

Recovery model: Simple

Compatibility level: SQL Server 2012 (110)

Containment type: None

Other options:

Auto Shrink	False
Auto Update Statistics	True
Auto Update Statistics Asynchronously	False
Containment	
Default Fulltext Language LCID	1033
Default Language	English
Nested Triggers Enabled	True
Transform Noise Words	False
Two Digit Year Cutoff	2049
Cursor	
Close Cursor on Commit Enabled	False
Default Cursor	GLOBAL
FILESTREAM	
FILESTREAM Directory Name	
FILESTREAM Non-Transacted Access	Off
Miscellaneous	
Allow Snapshot Isolation	False
ANSI NULL Default	False

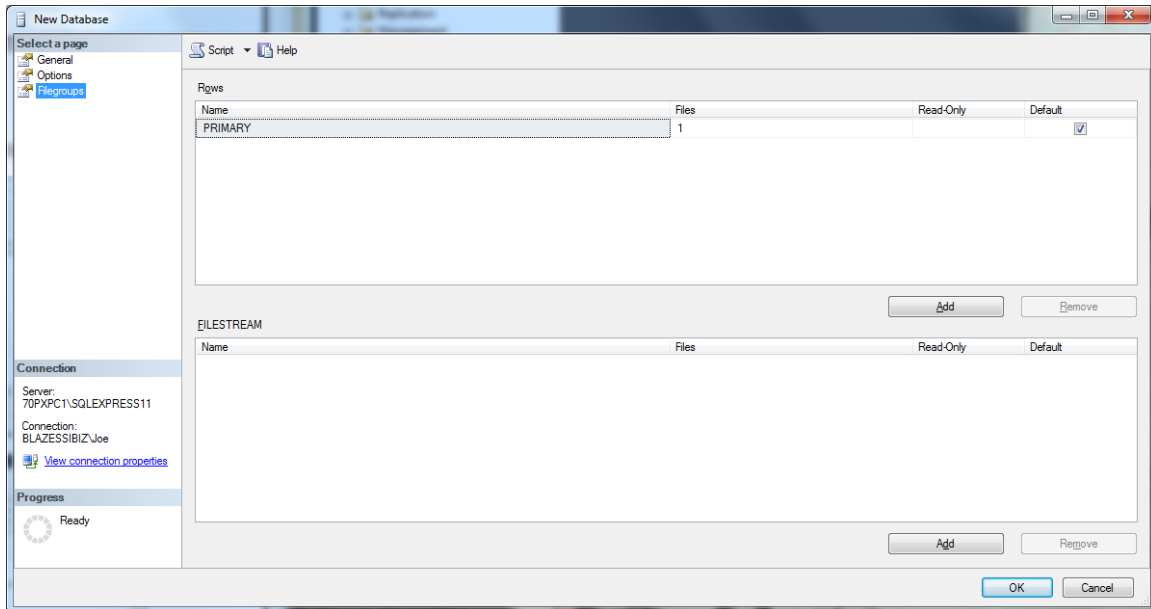
Allow Snapshot Isolation

Connection: Server: 70PXP1\SQLEXPRESS11
Connection: BLAZESSIBIZJoe
[View connection properties](#)

Progress: Ready

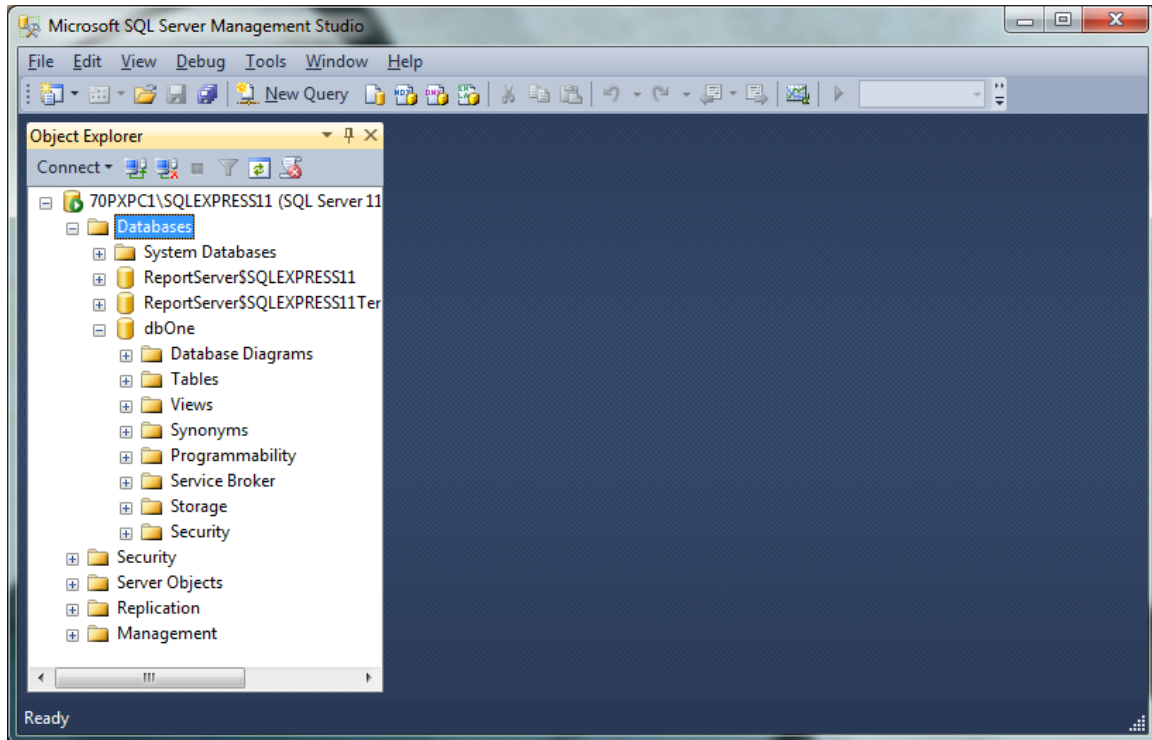
OK Cancel

Contact your database administrator to determine the appropriate values to enter on the Filegroups dialog:



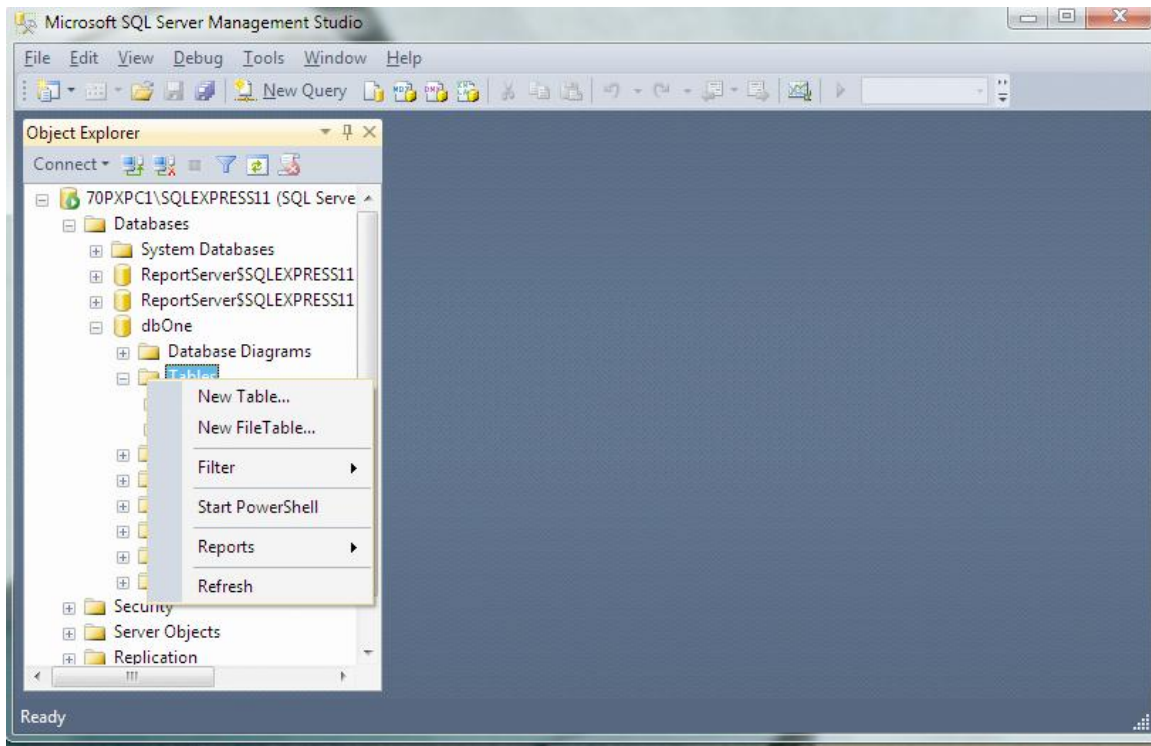
Click OK to create the new database.

The Management Studio should now reflect the new database. Open the dbOne database node to reveal the database elements:

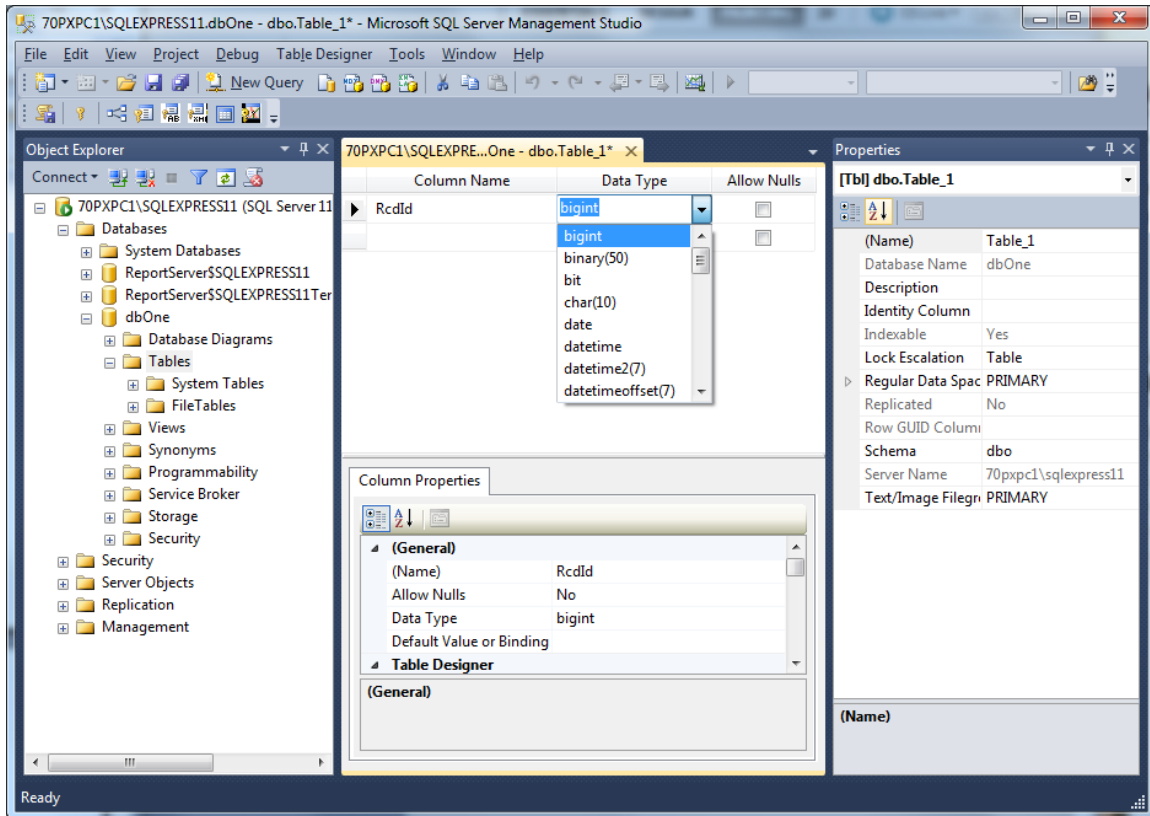


Adding a Table to a Database:

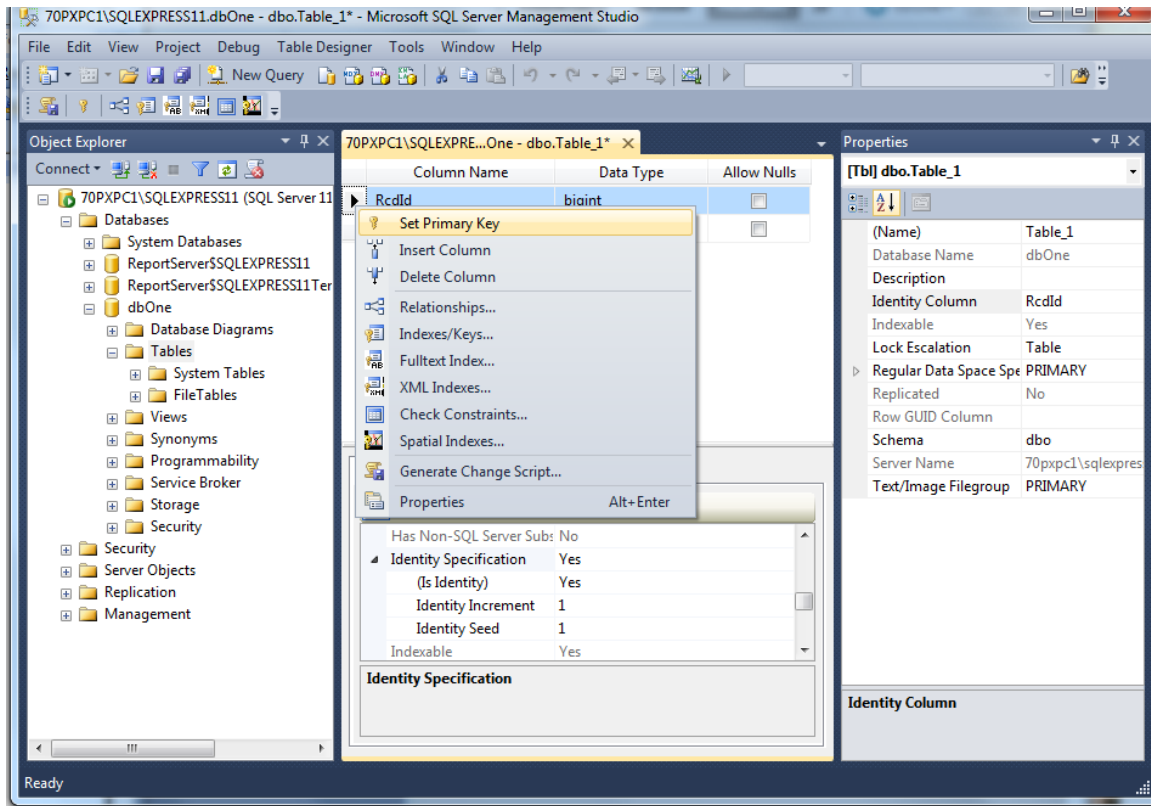
Right click the Tables node of the dbOne database node and select New Table...:



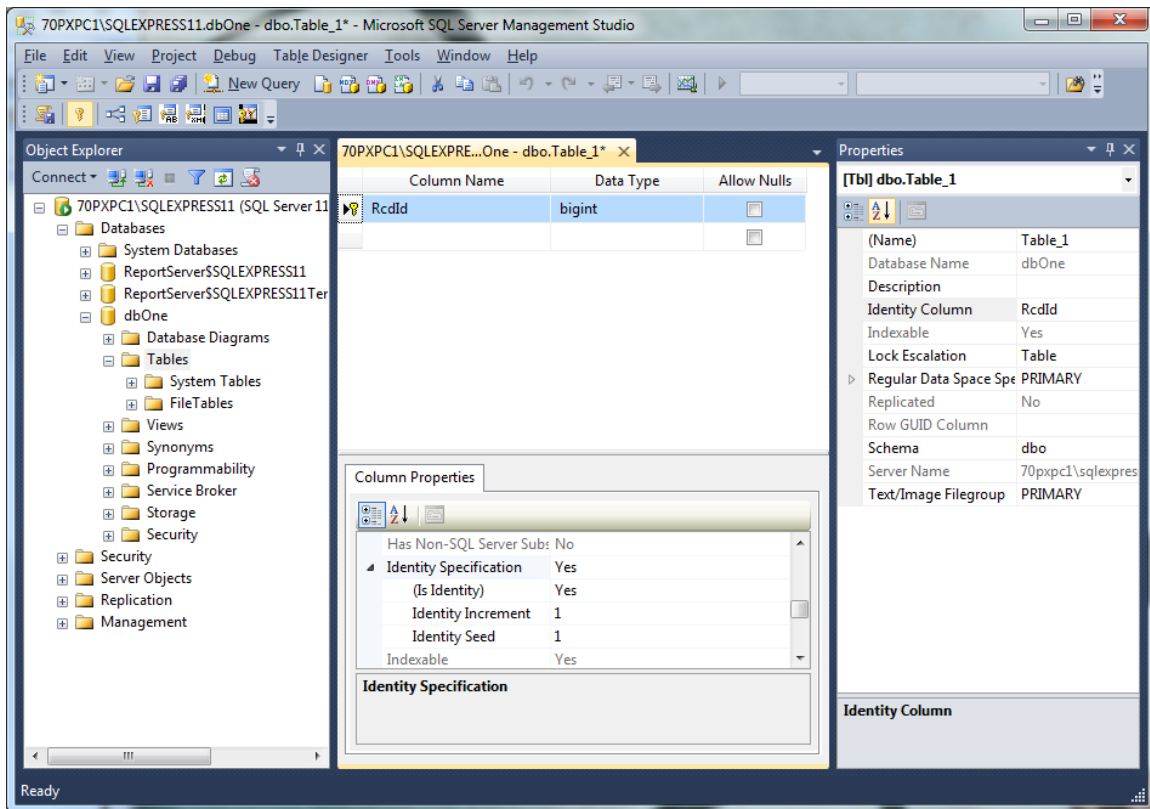
Add the first column (RcdId) as the key column for the table, not allowing nulls and with bigint data type:



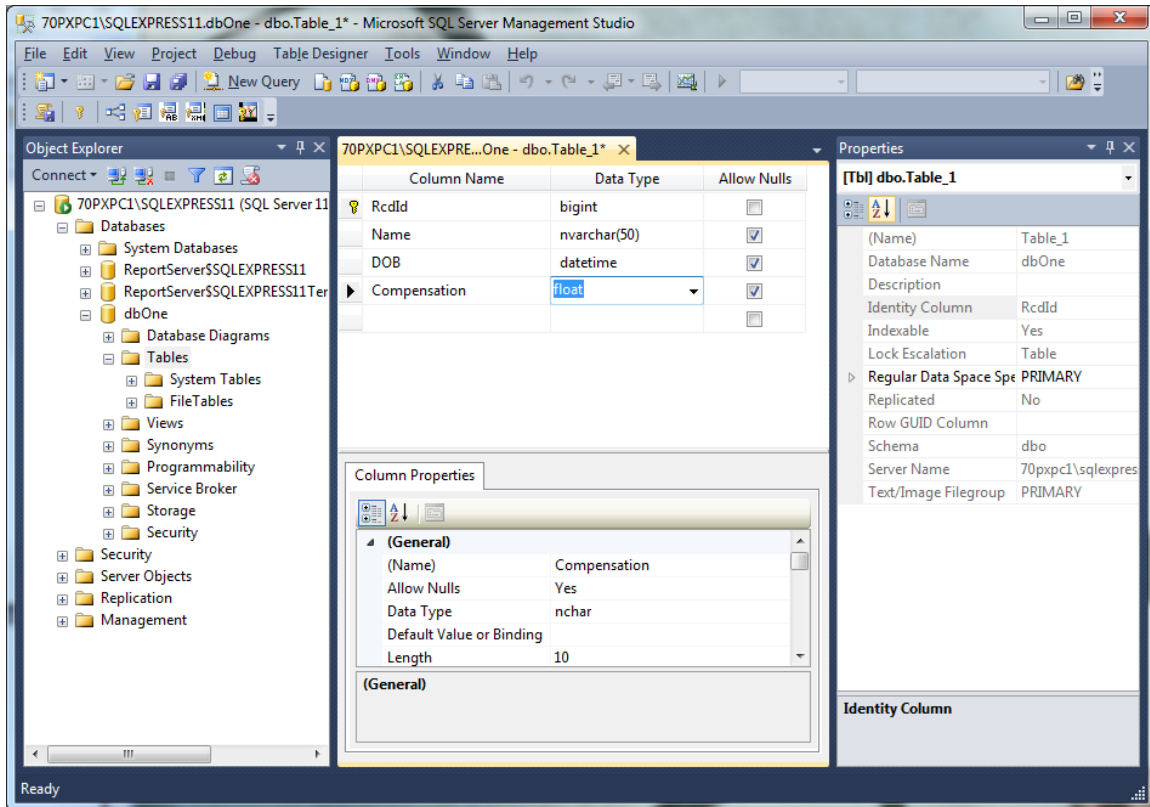
Right click the [>] to the left of the Column Name value (RcdId in this example) to set it as the primary key for this table:



Make this column an 'Identity Column'. Using the Column Properties tab, expand the Identity Specification section to select the Is Identity = Yes, Identity Increment = 1 and Identity Seed = 1:

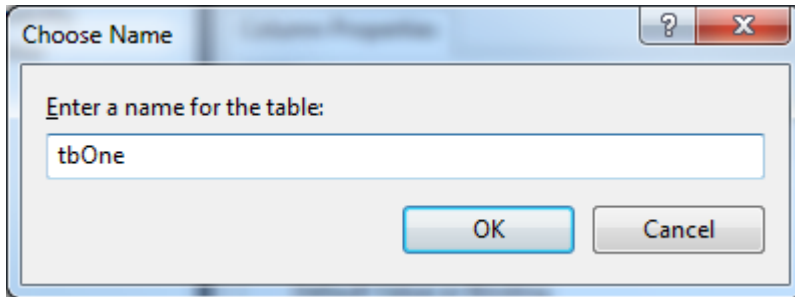


Using this table column definition region of the Management Studio, add the desired additional columns for this table (Name, DOB and Compensation):

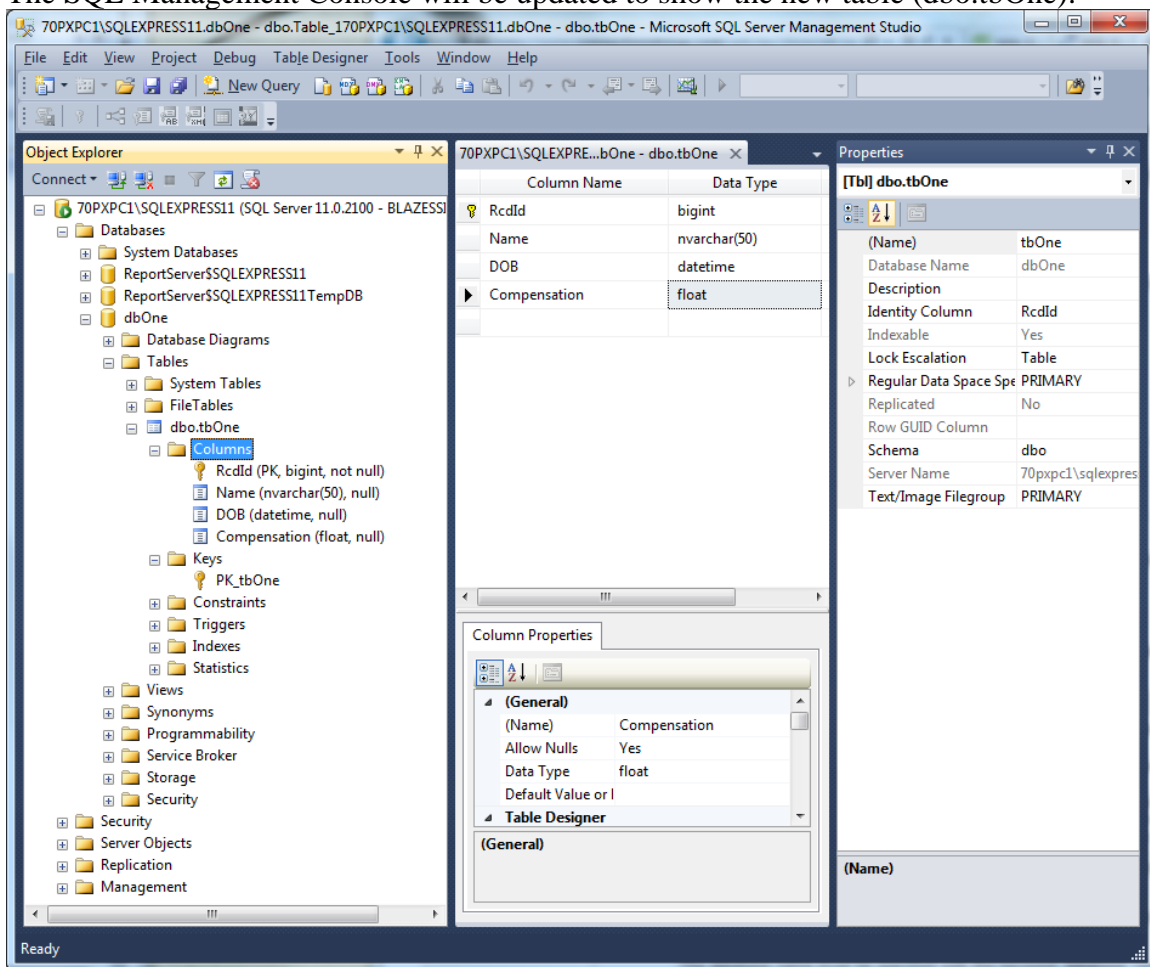


Additional columns may be added to the tbOne table, if desired, using the using the above-described process.

Finally save this table using File > Save All or File > Save Table_1 to present the Choose Name dialog. After entering the table name (tbOne), click the OK button:



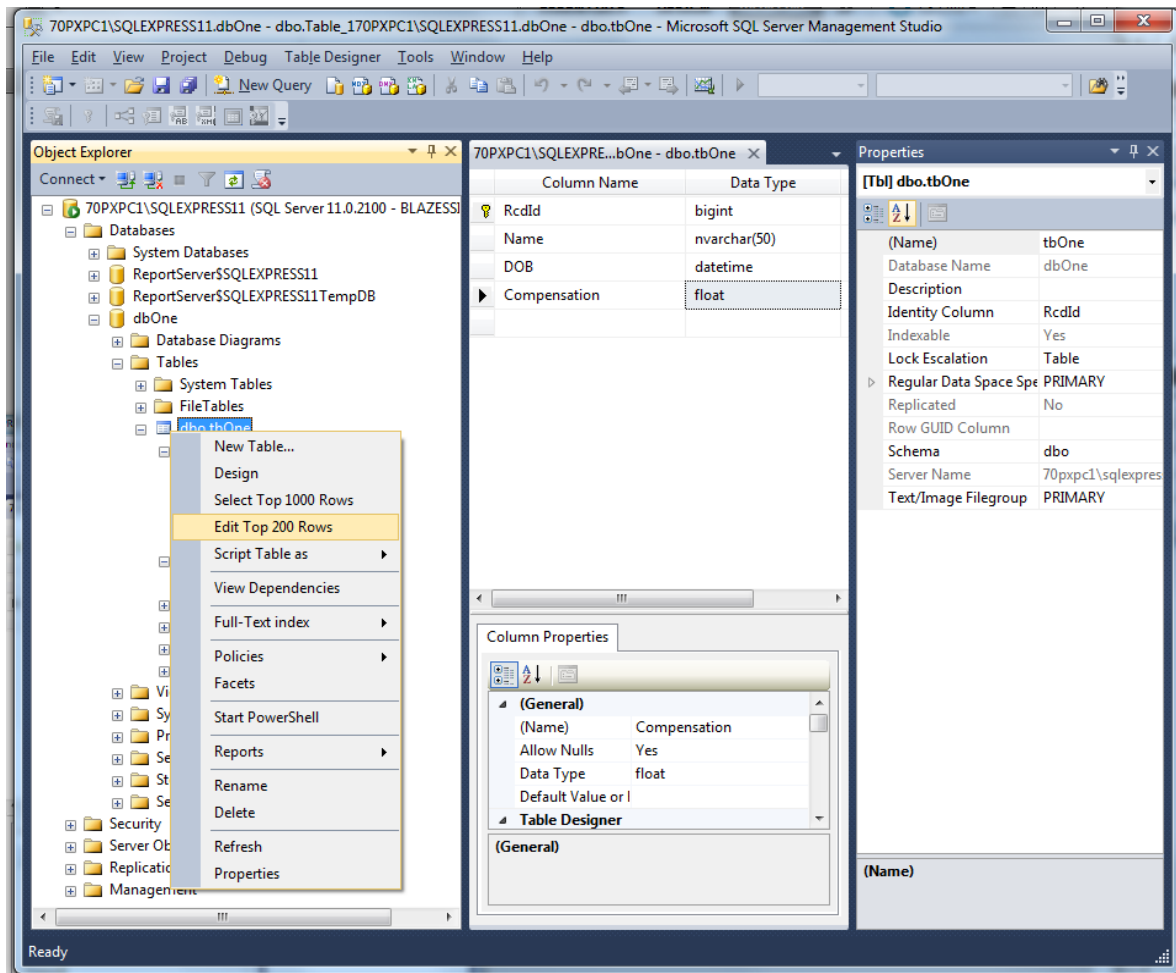
The SQL Management Console will be updated to show the new table (dbo.tbOne):



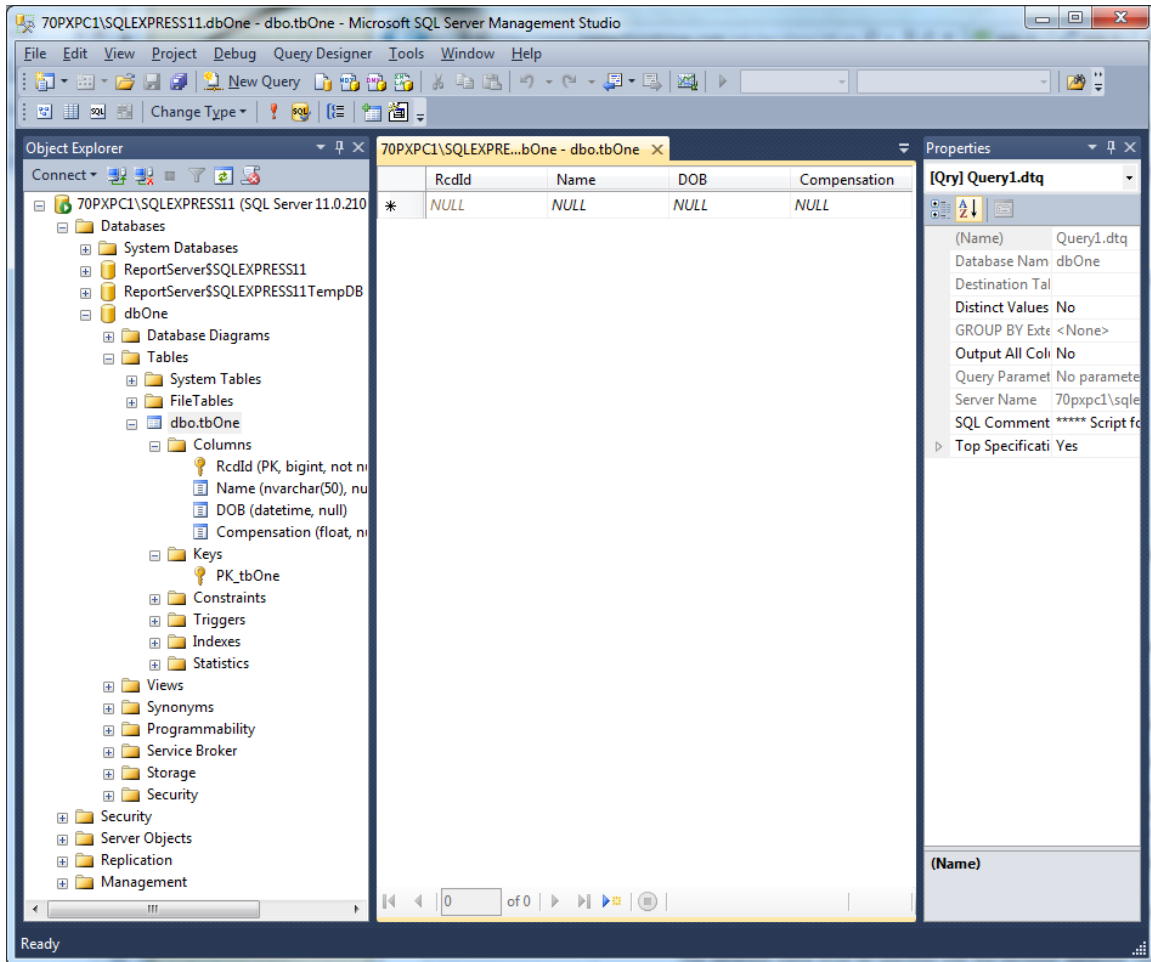
Add additional tables for this database, if desired, using the above-described process.

Manually Adding Data to an SQL Table:

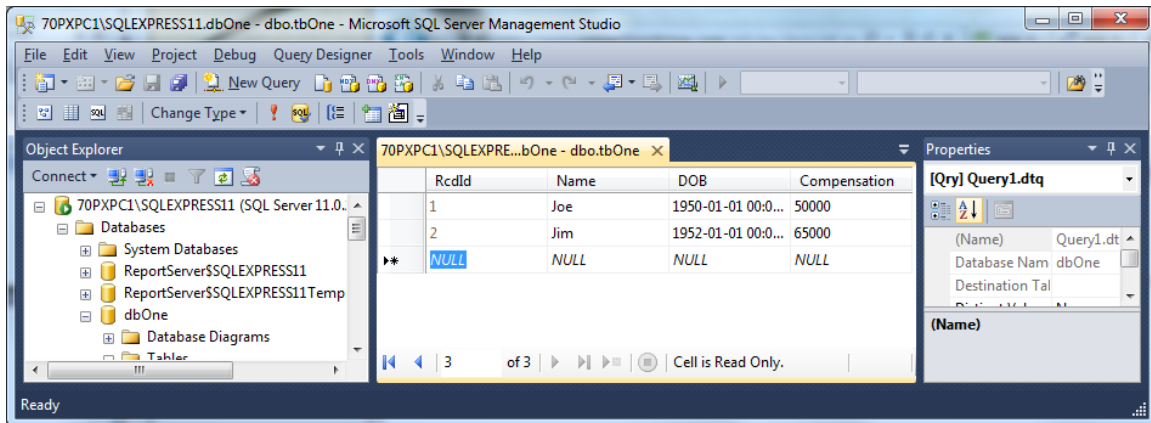
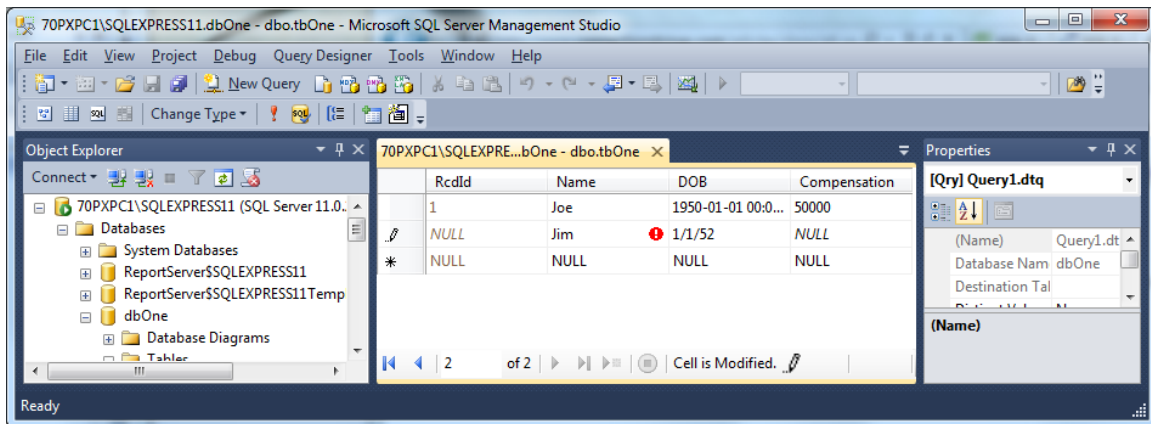
Right click the tbOne table node and select Edit Top 200 Rows option:



Enter the Name, DOB and Compensation column values for each record in the tbOne table:



The RcdId field will be automatically incremented when the record is saved to the tbOne table. Moving the focus to the new record row in the dialog saves the previously-entered table records:



Keep adding records to the table as desired.

Stored Procedures in a Data Base:

Stored procedures are methods which implement application-specific data base algorithms.

Refer to:

<http://msdn.microsoft.com/en-us/library/ms190782.aspx>

Using the Microsoft-provided procedure, enter the following sample stored procedure into the dbOne data base:

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:          APLNext Author
-- Create date: 4/27/2012
-- Description: Sample Stored Proc #1
-- =====
CREATE PROCEDURE spGetCompensationFromName
    -- Add the parameters for the stored procedure here
    @NameMatch nvarchar(50) = NULL
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    -- Insert statements for procedure here
    SELECT Compensation FROM tbOne WHERE Name=@NameMatch;
END
GO
```